AMENDMENTS TO THE CLAIMS

1. (Currently amended) A method of consolidating using a computer system to consolidate multiple configuration models using an automated process, ~~wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and include at least one rule having a constraint that references a non-ancestral family to the constraint,~~ the method comprising:

determining if a conflict exists between at least two of the configuration models, wherein the configuration models are organized in accordance with respective directed acyclic graphs, each configuration model includes at least one ancestor configuration model family and a child configuration model family below the ancestor family, a first conflicting configuration model comprises a configuration model that includes a release of a product that is not released in at least a second conflicting configuration model and the product is defined using the ancestor and child configuration model families;

extending the ancestor family of the product in the first conflicting configuration model to be compatible with second conflicting configuration model;

restricting child family in the first conflicting configuration model so that the child family is not released in the extension of the ancestor family;

combining the configuration models into a single, consolidated model that maintains a non-cyclic chain of dependencies among families and features of families for use in answering configuration questions. ~~, wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and at least one model includes a rule that causes a configuration conflict with another model.~~

2. (Original) The method of claim 1 further comprising:

detecting any inconsistencies between rules included in the consolidated model; and

attempting to resolve any detected inconsistencies.

3. (Currently amended) A computer system for consolidating multiple models, ~~wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and include at least one rule having a constraint that references a non-ancestral family to the constraint,~~ the system comprising:

a processor; and

a memory, coupled to the processor, having code stored therein and executable by the processor~~, the code comprising~~ for:

determining if a conflict exists between at least two of the configuration models, wherein the configuration models are organized in accordance with respective directed acyclic graphs, each configuration model includes at least one ancestor configuration model family and a child configuration model family below the ancestor family, a first conflicting configuration model comprises a configuration model that includes a release of a product that is not released in at least a second conflicting configuration model and the product is defined using the ancestor and child configuration model families;

extending the ancestor family of the product in the first conflicting configuration model to be compatible with second conflicting configuration model;

restricting child family in the first conflicting configuration model so that the child family is not released in the extension of the ancestor family;

~~a model consolidation module to combine~~ combining the configuration models into a single, consolidated model that maintains a non-cyclic chain of dependencies among families and features of families for use in answering configuration questions.~~, wherein each model comprises only rules that define a non-cyclic chain of~~

~~dependencies among families and features of families and at least one model includes a rule that causes a configuration conflict with another model.~~

4. (Currently amended) A computer program product having instructions encoded therein to consolidate multiple models, ~~wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and include at least one rule having a constraint that references a non-ancestral family to the constraint,~~ the instructions comprising code ~~to~~ for:

determining if a conflict exists between at least two of the configuration models, wherein the configuration models are organized in accordance with respective directed acyclic graphs, each configuration model includes at least one ancestor configuration model family and a child configuration model family below the ancestor family, a first conflicting configuration model comprises a configuration model that includes a release of a product that is not released in at least a second conflicting configuration model and the product is defined using the ancestor and child configuration model families;

extending the ancestor family of the product in the first conflicting configuration model to be compatible with second conflicting configuration model;

restricting child family in the first conflicting configuration model so that the child family is not released in the extension of the ancestor family;

~~combine~~ combining the configuration models into a single, consolidated model that maintains a non-cyclic chain of dependencies among families and features of families for use in answering configuration questions.~~, wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and at least one model includes a rule that causes a configuration conflict with another model.~~

5. (Currently amended) The method of claim 1 wherein the configuration models represent configuration models of vehicles.

6. (Previously Presented) The method of claim 1 wherein the consolidated model includes only buildable configurations.

7. (Currently amended) The method of claim 1 wherein:

~~combining the models into a single, consolidated model~~ extending the ancestor family of the product in the first conflicting configuration model to be compatible with second conflicting configuration model further comprises:

extending a rule from ~~one of the models~~ the first conflicting configuration model into [[an]] the ancestor ~~of a~~ family ~~of a defining constraint~~; and

restricting child family in the first conflicting configuration model so that the child family is not released in the extension of the ancestor family further comprises:

repairing the extension of the rule in ~~a child of the ancestor of the family of the defining constraint~~ the child family.

8. (Currently amended) The method of claim 1 wherein combining the models into a single, consolidated model further comprises:

loading the configuration models into a memory of the computer system;

constructing a directed acyclic graph of all rules in all the configuration models;

for each configuration model, determining which portions of an overall configuration space for which the configuration model does not provide a buildable configuration; and

for each configuration model, constraining statements of the rules ~~with in~~ within the configuration model to fall within a space of defining features of the configuration model[[;]].

9. (Currently amended) The method of claim 8 wherein determining which portions of an overall configuration space for which each configuration model does not provide a buildable configuration further comprises:

determining which families are ancestors of families of defining constraints; and

subtracting a right hand side and a left hand side of each rule of each family that are ancestors of families of defining constraints from a rule representing all buildable configurations.

10. (Currently amended) The system of claim 3 further comprising code [[to]] for:

~~detect~~ detecting any inconsistencies between rules included in the consolidated model; and

~~attempt~~ attempting to resolve any detected inconsistencies.

11. (Currently amended) The system of claim 3 wherein the configuration models represent configuration models of vehicles.

12. (Previously Presented) The system of claim 3 wherein the consolidated model includes only buildable configurations.

13. (Currently amended) The system of claim 3 further comprising code [[to]] for:

~~extend~~ extending a rule from ~~one of the models~~ the first conflicting configuration model into [[an]] the ancestor ~~of a~~ family ~~of a defining constraint~~; and

~~repair~~ repairing the extension of the rule in ~~a child of the ancestor of the family of the defining constraint~~ the child family.

14. (Currently amended) The system of claim 3 further comprising code [[to]] for:

~~load~~ loading the configuration models into a memory of the computer system;

~~construct~~ constructing a directed acyclic graph of all rules in all the configuration models;

for each configuration model, ~~determine~~ determining which portions of an overall configuration space for which the configuration model does not provide a buildable configuration; and

for each configuration model, ~~constrain~~ constraining statements of the rules ~~with in~~ within the configuration model to fall within a space of defining features of the configuration model[[;]].

15. (Currently amended) The system of claim 14 further comprising code [[to]] for:

~~determine~~ determining which families are ancestors of families of defining constraints; and

~~subtract~~ subtracting a right hand side and a left hand side of each rule of each family that are ancestors of families of defining constraints from a rule representing all buildable configurations.

16. (Currently amended) The computer program product of claim 4 further comprising code [[to]] for:

~~detect~~ detecting any inconsistencies between rules included in the consolidated model; and

~~attempt~~ attempting to resolve any detected inconsistencies.

17. (Previously Presented) The computer program product of claim 4 wherein the models represent configuration models of vehicles.

18. (Currently amended) The computer program product of claim 4 wherein the configuration models represent configuration models of vehicles.

19. (Currently amended) The computer program product of claim 4 further comprising code [[to]] for:

~~extend~~ extending a rule from ~~one of the models~~ the first conflicting configuration model into [[an]] the ancestor ~~of a~~ family ~~of a defining constraint~~; and

~~repair~~ repairing the extension of the rule in ~~a child of the ancestor of the family of the defining constraint~~ the child family.

20. (Currently amended) The computer program product of claim 4 further comprising code [[to]] for:

~~load~~ loading the configuration models into a memory of the computer system;

~~construct~~ constructing a directed acyclic graph of all rules in all the configuration models;

for each configuration model, ~~determine~~ determining which portions of an overall configuration space for which the configuration model does not provide a buildable configuration; and

for each configuration model, ~~constrain~~ constraining statements of the rules ~~with in~~ within the configuration model to fall within a space of defining features of the configuration model[[;]].

21. (Currently amended) The computer program product of claim 20 further comprising code [[to]] for:

~~determine~~ determining which families are ancestors of families of defining constraints; and

~~subtract~~ subtracting a right hand side and a left hand side of each rule of each family that are ancestors of families of defining constraints from a rule representing all buildable configurations.

22. (Currently amended) A computer system for performing an automatic consolidation of multiple models of configurable products, the system comprising:

means for determining if a conflict exists between at least two of the configuration models, wherein the configuration models are organized in

accordance with respective directed acyclic graphs, each configuration model includes at least one ancestor configuration model family and a child configuration model family below the ancestor family, a first conflicting configuration model comprises a configuration model that includes a release of a product that is not released in at least a second conflicting configuration model and the product is defined using the ancestor and child configuration model families;

means for extending the ancestor family of the product in the first conflicting configuration model to be compatible with second conflicting configuration model;

means for restricting child family in the first conflicting configuration model so that the child family is not released in the extension of the ancestor family;

means for combining the configuration models into a single, consolidated model that maintains a non-cyclic chain of dependencies among families and features of families for use in answering configuration questions.~~, wherein each model comprises only rules that define a non-cyclic chain of dependencies among families and features of families and at least one model includes a rule that causes a configuration conflict with another model.~~

REMARKS

Claims 1-22 are pending.

Claims 1-22 stand rejected.

Claims 1, 3-5, 7-11, 13-16, and 19-22 have been amended.

Claim Objections

Claims 8, 14 and 20 are objected to for improper punctuation and have been amended to end with a period instead of a semicolon.

Applicants respectfully request withdrawal of the rejection.